大略看过 作用不大

A Context-Aware Trust-Based Information Dissemination Framework for Vehicular Networks

Karim Rostamzadeh, *Student Member, IEEE*, Hasen Nicanfar, *Student Member, IEEE*, Narjes Torabi, *Student Member, IEEE*, Sathish Gopalakrishnan, *Member, IEEE*, and Victor C. M. Leung, *Fellow, IEEE*

***Abstract*—Reliable, secure, private, and fast communication in vehicular networks is extremely challenging due to the highly mobile nature of these networks. Contact time between vehicles is very limited and topology is constantly changing. Trusted communication in vehicular networks is of crucial importance because without trust, all efforts for minimizing the delay or maximizing the reliability could be voided. In this paper, we propose a trust-based framework for a safe and reliable information dissemination in vehicular networks. The proposed framework consists of two modules such that the first one applies three security checks to make sure the message is trusted. It assigns a trust value to each road segment and one to each neighborhood, instead of each car. Thus, it scales up easily and is completely distributed. Once a message is evaluated and considered to be trustworthy, our method then in the second module looks for a safe path through which the message is forwarded. Our frameworks are application-centric; in particular, it is capable of preserving traffic requirements specified by each application. Experimental results demonstrate that this framework outperforms other well-known routing protocols since it routes the messages via trusted vehicles.**

***Index Terms*—Information dissemination, security, trust, vehicular networks (VNets).**

I. Introduction

VEHICULAR networks (VNets) are the basis for different applications offered in intelligent transportation systems (ITS). These envisioned applications cover a large spectrum ranging from delay-sensitive safety-related applications such as accident notifications, to delay-tolerant services such as file sharing. No matter what the application is, routing or in general, information dissemination in a vehicular network is challenging due to the highly dynamic and mobile nature of these networks [1]. Different applications in VNets can be offered using either broadcasting the information, which is the case mainly in safety-related applications, or unicasting/multicasting, which happens mostly in nonsafety applications.

Over the past few years, researchers have proposed several broadcasting and unicasting routing protocols for VNets that minimize the delivery delay or the communication interference [2]–[4]. However, the missing key point in those methods is the security analysis. A node is assumed to be malicious when it manipulates or intentionally drops a message. Consider this simple example that explains the significance of the problem: suppose one vehicle has a message and it has two neighbors from which it chooses to forward its message. One of the neighbors is a trustworthy node that relays every message without any problem; however, routing from that node increases the delay. The other neighbor, on the other hand, is on a fast route to the destination but it is malicious and drops the messages half the time. Now, based on a normal routing protocol that its only focus is on minimizing the delay, the malicious node will be selected to route the message faster. However, there is a chance of 50% that the message will be dropped at this node and it never reaches the destination. So, ignoring malicious nodes can void the whole algorithm efforts to minimize the delay. This example clearly shows why there is a need for a safe and reliable communication framework in VNets. Note that our focus is on having a secure packet delivery in addition to a reliable one. Packet reliability, a.k.a. packet delivery ratio, could be damaged by environment and network characteristics such as channel fading or collisions. Secure packet delivery, on the other hand, can be hurt because of attacks and intentional manipulations by malicious nodes.

Trust is a human and psychological factor that is historically very well known in the social interactions. Using artificial intelligence (AI) and bringing the concept of the trust to the information and communication technology (ICT) has been proposed and studied during the last decade [5]. For instance, trust of a relay node i can be defined as the number of packets that i has relayed without manipulating them, out of the total number of packets given to the i to be relayed. Then, the probability of i relaying a new packet is the trust value of i based on this simple rule: *A node acts the same way that it has done so far.* As our first contribution, we leverage this concept to cover a range of application requirements such as security, privacy, delay, reliability to name a few. There is a trust value calculated for each of these parameters. Then, we develop a function of all these trust values to calculate the total trust for each path, which is our reference for choosing the best route.

比例关系

Trust in a VNet faces two main challenges: 1) the speed of vehicles which changes the topology constantly and minimizes the contact times between the nodes and 2) the lack of a centralized third party to evaluate and maintain the trust values. Majority of the trust models proposed for VNets are entity-centric in which the focus is on verifying the vehicle credentials [6]. Once the source is authenticated, the message can be trusted. There are a few data-centric approaches that focus on the correctness of the received message, instead [7]. Both

Manuscript received October 21, 2014; revised December 02, 2014; accepted December 12, 2014. Date of publication January 07, 2015; date of current version March 13, 2015. This work was supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada.

The authors are with the Department of Electrical and Computer Engineering, University of British Columbia, Vancouver BC V6T 1Z4, Canada (e-mail: karimr@ece.ubc.ca; hasennic@ece.ubc.ca; ntorabi@ece.ubc.ca; sathish@ece.ubc.ca; vleung@ece.ubc.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JIOT.2015.2388581

of these models suffer from scalability and the entity-centric models have the assumption that there is always a third-party certificate issuer in the vicinity that is often not valid in the context of VNets.

We survey the state-of-the-art in trust models/systems and routing protocols in VNets. Then, as our next contribution, we propose a novel two-layer framework for application-oriented context-aware trust-based communication (FACT) in VNets, where nodes only use their most trusted neighbors to forward the message otherwise they carry the message by themselves. FACT consists of two modules: *Admission* and *Dissemination*. The key distinction of the FACT lies in its *space-centric* nature. It is a combination of entity-/data-centric methods in addition to its focus on location. Once a message is received, FACT first applies three safety checks in the admission module to make sure the message: 1) originated from a trusted region and traversed a trusted path; 2) was not under attack on its path; and 3) has a valid content. Then, FACT admits the message and pushes it to the dissemination module to be forwarded through a trusted path. Each vehicle has a trust table where each road segment in the city has its own trust value and this value is constantly updated by the vehicle based on its experience in that segment. The intuition behind FACT is that some areas of every city are known to be safer with better facilities. It is fair to assume that vehicles in those areas are more trusted, as well. Otherwise and when there are malicious vehicles in the area, that neighborhood's trust is reduced and it gets a bad reputation. We divide vehicular networks' applications into three main categories and then identify major requirements of each category. FACT is a general admission/communication framework that accommodates all requirements of each application while making sure the connection is trusted. It supports both broadcasting and unicasting/multicasting modes. After explaining the details of FACT, we evaluate our framework via simulation and show that FACT outperforms other routing protocols when some areas of the city are not trusted.

FACT gives network designers a full package, which delivers trustworthy messages through a safe path with high reliability and in a short amount of time. It is flexible enough meaning network admins can tune the parameters based on the network condition. Designers can incorporate their desired scheduling and routing schemes into FACT and still disseminate the messages safely. FACT is a framework that supports different applications with different requirements.

II. Related Work

Secure routing, privacy, and trust in VNets have gained attention from the research community over the past few years. However, private communication is still in its early days, and routing remains challenging. One of the first yet important routing protocols for VNets is epidemic routing [2], in which each packet is replicated until reaches a certain destination. A carry-and-forward strategy is used in vehicle-assisted data delivery in vehicular ad hoc networks (VADD) [3], where messages can be carried by vehicles in sparse networks when there are no neighbors to which it can be forwarded. VADD always tries to use vehicles on the fastest roads available for forwarding the message. However, vehicles may deviate from predicted routes and the routing path should be continuously recalculated. D-Greedy and D-MinCost are two unicast forwarding methods proposed by Skordylis and Trigoni [4]. Unlike VADD, they tried to minimize the number of transmissions instead of focusing on minimizing the delivery delay.

Kamvar *et al.* [8] concentrate on trust in the peer-to-peer (P2P) file-sharing networks, based on the peers' history of uploads, called EigenTrust. They proposed an algorithm aiming at decreasing the number of downloads of inauthentic files in the network that assigns each peer a unique global trust value. They also described a distributed and secure method to compute global trust values, based on power iteration. In fact, peers use these values to choose from whom they download, whereas the network identifies malicious peers and isolates them from the network. This proposal mainly concentrates on P2P and directly downloading the files. However, in a dynamic environment, such as vehicular ad hoc network (VANET), with a high mobility, using the concepts such as score manager is not feasible.

The framework presented in [9] aimed at dealing with potentially untrustworthy information. The framework includes a computational trust model for estimating the amount of received information uncertainty. In order to realize the drivers goals, the framework also has a probabilistic beliefs-desires-intentions agent system for reasoning about this uncertain information. In order to secure, DTN routing toward efficient trust establishment, iTrust scheme proposed by Zhu *et al.* [10], as probabilistic misbehavior detection. They used a trusted authority (TA) to judge the nodes behavior, which is periodically available, based on the collected routing evidences and probabilistically checking. For the TA to ensure the security of DTN routing at a reduced cost, they modeled the inspection game and use game theoretical analysis to demonstrate that by setting an appropriate investigation probability. Also to improve the efficiency of the proposed scheme, they allows a dynamic detection probability determined by the trust of users by correlating detection probability with the reputation of that node.

The proposed mechanism in [11] aimed at *ad hoc* environment by designing an ad hoc on-demand distance vector routing (AODV)-based routing protocol that combines the hop count and trust value. When a source studies different path opportunities, it multiplies the trust value of each node to obtain the trust value of the path. Although it may be true in the packet delivery ratio, the trust value of the path cannot be always calculated by a multiplication of the trust value of the nodes along the path, e.g., in anonymity-based one. Moreover, the proposed routing protocol in [12] is trust-based aimed at mobile ad hoc networking (MANET) that concentrates on energy efficiency.

Situation-aware trust (SAT) [6], a trust architecture for VNets, reveals the identity of the users by using a social network. The trust system proposed in [7], designed for the receivers and broadcast communication, shifts the concentration from the sender to the data itself. In fact, the content of the message defines the trust value even the message is generated by different vehicles that have different trust levels. The Markov chain-based trust model for VNets presented in [13] is a hybrid model evaluating trust based on the data and the entity together, in order to filter out malicious and selfish nodes. Aside from the complexity of the proposed method, it is not clear how this algorithm works when there is no connectivity in the network or when the application needs a set of delay-reliability requirements.

The aim of a trust and reputation infrastructure-based proposal (TRIP) for vehicular ad hoc networks [14] is broadcast communication in VNet. It is designed for a receiver evaluating the trustworthiness of the received message, while the message can be accident (safety) or a weather condition (nonsafety). Their guideline for designing a trust system in VNet identifies five characteristics such as: simple, light, and fast; accurate; scalable; resilient to security and privacy threats; and not dependent on mobility patterns. The scope of A vehicle ad hoc network reputation system (VARS) [15] is also broadcasting messages. A relay node in forwarding a message, generates its own opinion based on the sender if it is known, or based on similar received messages in the past, and sends the opinion along with the message. However, considering dynamic nature of VNets, VARS mainly relies on others opinion along with indirect trust values for a receiver on trusting a message, which are not enough trusted and valid sources for the receiver. Moreover, the proposed trust model in [16] concentrates on broadcast communication of safety related messages where is limited to utilizing the signature of a message issued by the originator to preserve trust of the messages.

The trust model proposed by Wang *et al.* [17] is designed for senders. A sender finds similar vehicles to forward the message. Finding similar nodes based on location, energy, and brand is not practical and does not guarantee of being a better candidate: the same brand does not make any difference; a node with the same energy level cannot be a good one, if the sender is frustrating of low energy where requires a neighbor with a higher energy; and energy is not a case in the VNets at all.

In a highway and reporting the traffic congestion, a malicious vehicle may send bogus congestion information to others, which is the focus of Huang *et al.* [18]. In their design, they leveraged traffic fellow theory to observe the Kinematic wave caused by congestion. So without a need to a centralized controlled congestion detection and prediction system, each vehicle only relied on local speed and distance measurements to validate the congestion event that is sent by another vehicle. In [19], an road side unit (RSU) and beacon-based trust management system are proposed that aimed at broadcasting message opinions quickly. Therefore, and in a privacy-enhanced VANET environment, if an internal attacker sends or forwards forged messages, the message will be foiled. Eiza and Ni [20] leveraged location and velocity information of vehicles in order to calculate link reliability for their reliability-based routing scheme for VANETs. Their objective was routing process in which they tried to facilitate quality of service (QoS) support in it.

In order to receive trusted traffic information by a vehicle, Teler and Cristea [21] designed a trust-based security system, where regardless of source of the data, the trust is measured for individual pieces of data referring to a specific event. In this model, a driver evaluated the trust of received information, and then disseminates the trust value to other vehicles in order to improve the accuracy in the trustworthiness of an event.

In [22], a decentralized lightweight authentication scheme for vehicle-to-vehicle communication networks is proposed, which is called trust-extended authentication mechanism. In fact, they aimed at improving the performance of the authentication procedure by using the concept of transitive trust relationships. Gazdar *et al.* [23] used Markov to formalize variation and its stability of the trust metric in their trust model for VANET. They considered the dynamic nature of the trust metric due to the vehicle behavior. In addition, they brought into account constraints related to the monitoring process. In their model, each vehicle monitors and updates the trust metric of its neighbors based on the behavior of the neighbors.

In [24], Haddadou and Rachedi proposed DTM^2 aimed at issue of malicious and selfish nodes in the VANET. They used the concept of cost, in which they assumed a sender transmits a signal, which has a cost based on the message and sender behavior, with its message to show a guarantee of the truthfulness of the message to the potential receivers. In this paradigm, the cost is calculated in a way that the cost increases when the behavior of the sender gets worse. Therefore, the cost for the malicious and selfish nodes increases eventually, as a punishment. Similarly, the cost for the benign nodes decreases as a reward proportionally to the signals value. In [25], Liao *et al.* brought into account the trustworthiness of the message originator and the forwarding nodes, if any, to calculate the trust of the message itself, in a vehicle-to-vehicle (V2V) communication. They used the existing vehicle-to-infrastructure (V2I) communication facilities deployed and managed by central traffic authorities for collection of the vehicle behavior in a crowd-sourcing fashion.

III. Proposed Trust Model

In this section, first we describe some of the definitions for our framework. Then, calculations regarding trust value and the total trust are presented to be used in our design as part of Section IV.

The level of performing the expected service by a trustee is called *satisfaction*, which in fact derived the trustworthiness. We assume the trust value to be a number between (0,1). The four major parts of the trust system are *initial trust, trust metric, operation, and trust management*. Identifying the appropriate trust metric, which is relevant to the application, and a mechanism to measure the metric, e.g., a binary or percentage unit, are part of a trust system. In addition, designing a system that manages the trust is needed, which can be centralized, distributed, or a combination of them.

Definition 1: Trust assessment can be direct or indirect. In direct assessment, a trustor relies on its own experience about a trustee. For instance, in a wireless communication environment, if node S (trustor) sends a packet to node R (trustee), which is one hop away, to forward (service) the packet to node D, which is two hops away, S can overhear R, and observes if R is genuinely sending the packet. On the other hand and in indirect assessment, a trustor relies on other nodes experience of dealing with a trustee.

FACT is based on carry-and-forward information dissemination. Therefore, if there is no vehicle in the vicinity, the vehicle

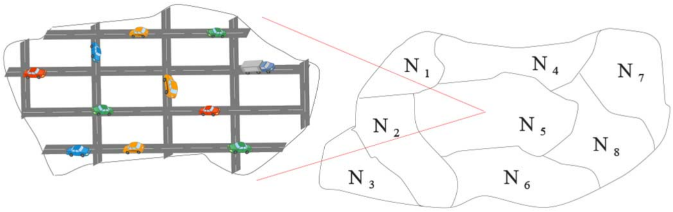


Fig. 1. Right: map of the city divided to different neighborhoods. Left: each neighborhood consists of different road segments.

carrying the message should continue carrying the message until it comes across another vehicle and then decide on forwarding the message. The rest of this section is based on the assumption of having at least one vehicle in the transmission range. FACT is a general framework that allows messages of different categories of traffic to be securely delivered to their destinations based on their requirements. As shown in Fig. 1, all vehicles maintain a trust map, which divides the city into several neighborhoods and each neighborhood itself is divided to multiple segments. A segment is part of the road between two intersections. Our work is not limited to a specific sectoring method; it can be symmetrical or asymmetrical sectoring. Map division should be done in a way that meets the network administrator's resolution requirement without imposing excessive overhead. In the mentioned trust map and for every segment, a trust value TR is recorded for each time of the day in each specific day of the week. For simplicity, we only consider rush hour and regular hour for one day, i.e., two entries per day and 14 entries in total per week for each segment. It is possible to further simplify and only differentiate between weekdays and weekends, which lowers the number of entities for a segment to four. Every vehicle starts with an initial trust value of 0.5 for each neighborhood. Then, it gradually updates different entries of the trust map for that segment/neighborhood based on its experience. For example, if this vehicle experiences a trustworthy communication in that segment during rush hour in weekdays, then it increases the corresponding trust value for those times and days and makes it close to 1. Whenever a vehicle is calculating the trust value of a road segment for the first time, it uses the neighborhood's trust instead. The details of the trust model and the updates are discussed in Section III-A.

Note that the key component of our trust framework is that instead of assigning a trust value to each node, which is not scalable and hard to maintain and even inefficient, we assign a trust value to a neighborhood and to each segment of that neighborhood. In other words, our main contribution is that FACT is space-centric, i.e., it uses location information to evaluate the trustworthiness of messages. There is a similar scenario in a real city where some neighborhoods are safer than the others. This is the intuition behind FACT. Note that there is no predefined trust values for different neighborhoods because all cars start the FACT by assigning the same value of 0.5 to all neighborhoods and then they gradually update their database as they learn more about different areas. However, in different times of the day, cars from other parts of the city may travel to these areas and lower the actual trustworthiness of these areas. That is why FACT assigns different trust values to different times of the day and constantly modifies these values. The biggest advantage of FACT is its scalability and low complexity even for a metropolitan area. The other advantage is its efficiency. In approaches, such as [6] and [26] where there is a trust value per node, resources are often wasted because we never come across that car again, but it is much more likely that we travel to the same area more than once.

A. Trust Calculation

In this paper, trust is assumed to have multiple dimensions based on the service requirements of the corresponding application, e.g., delay, reliability, security/confidentiality, privacy/anonymity to name a few. FACT guarantees security and offers QoS based on the application requirements. Different applications on a VNet can be divided to three general categories based on their specific set of requirements. 1) Cat_a: safety-related services such as accident, high traffic, and emergency brake notifications, and critical services when there is a disastrous situation like earthquake in place. In this category, delay, end-to-end reliability, and data integrity are important. Some safety-related applications may be more tolerant to delay than others like when a road condition needs to be reported to incoming cars. As for data integrity, receivers must be assured that the information based on which they are making decision is genuine. 2) Cat_b: infotainment such as parking availability and gas price notifications. Reliability, access control, source anonymity, and integrity are the main parameters to be considered in this category. Access control assures that only appropriate pairs can communicate to each other since each piece of information is destined for a specific vehicle or a set of vehicles. Source authentication along with access control helps vehicles trust their communication by making sure information is coming from a reliable source. 3) Cat_c: third-party services such as when taxis working for the same company want to share passenger availability information. Applications in this category need source authentication to guarantee that their information is from the right person, and reliability to maximize number of cars receiving the information.

The overall trust value for a given path m is defined as

$$t\mathrm{TR}_m = F_{\mathrm{TR}}((\alpha_1, \mathrm{TR}_1), (\alpha_2, \mathrm{TR}_2), \ldots, (\alpha_k, \mathrm{TR}_k)) \quad (1)$$

where F_{TR} is a function that combines the trust dimensions TR_i to obtain the total trust value of $t\mathrm{TR}$. Note that the value of α_i represents the weight of the TR_i, e.g., delay, in the calculation of the total trust driven by the application. Given the category of the traffic as mentioned above, FACT assigns appropriate α_i's and calculates the overall trust for a path. One simple example of the function is a linear combination by having $\alpha_i = 1/k$ for $\forall i$. One other example is having 0 for the nonimportant α_i's.

The model in [27] is used for calculating trust between agents in a multiagent environment. TR_i is defined as a function of its current and historical values as follows:

$$(\mathrm{TR}^{\tau}_{v})_i = \eta \times (\mathrm{TR}_{\mathrm{cur}})_i + (1 - \eta) \times (\mathrm{TR}^{\tau}_{v-1})_i \quad (2)$$

where τ is the period of time (e.g., an hour, one day, or one week) and n is the number of the experiments that are considered so far. The value of η $(0 \leq \eta \leq 1)$ is used to give weight to the trust value of current period $(\mathrm{TR}_{\mathrm{cur}})_i$, comparing to last $v-1$ periods $((\mathrm{TR}^{\tau}_{v-1})_i)$ of the system in final trust calculation $((\mathrm{TR}^{\tau}_{v})_i)$. Since we only rely on the direct trust calculation in this model, the current trust value $(\mathrm{TR}_{\mathrm{cur}})_i$ is equal to the final *satisfaction* level of current period as per (3)

$$(\mathrm{TR}_{\mathrm{cur}})_i = \mathrm{ST}^{\tau}_{n} \tag{3}$$

where ST^{τ}_{n} is the satisfaction value of the trustee calculated by a trustor and is defined as

$$\mathrm{ST}^{\tau}_{n} = \gamma \times \mathrm{ST}_{\mathrm{cur}} + (1-\gamma) \times \mathrm{ST}^{\tau}_{n-1} \quad \mathrm{ST}^{\tau}_{0} = 0.5 \tag{4}$$

where γ $(0 \leq \gamma \leq 1)$ represents the weight that is given to the current experiment $(\mathrm{ST}_{\mathrm{cur}})$ comparing to the last experiments in this period $(\mathrm{ST}^{\tau}_{n-1})$ to calculate the final satisfaction value (ST^{τ}_{n}).

Satisfaction $(\mathrm{ST}^{\tau}_{n}$ or $\mathrm{ST}_{\mathrm{cur}})$ is a value between zero and one, where one means fully satisfied and zero means completely unsatisfied. We use the beta function to calculate the satisfaction value via (5), as it is the most referred function in the literature

$$\mathrm{ST}_{\mathrm{cur}} = \frac{N_{\mathrm{suc}} + 1}{N_{\mathrm{fail}} + N_{\mathrm{suc}} + 2} \tag{5}$$

where $\mathrm{ST}_{\mathrm{cur}}$ is the satisfaction level based on the number of successful (N_{suc}) and failed (N_{fail}) operations/services in an experiment, which consists of one or multiple packet(s) delivery. For instance, in a safety application, there may be only one packet while in the online social networking, the experiment consists of multiple packets.

Finally, every vehicle uses the total trust on each path as the reward of the path (1/cost) to choose the best path. In other words, FACT maximizes the trust value and selects the path that delivers the maximum trust. Let Set P denote the set of l paths where each one has a total trust value of $t\mathrm{TR}_m$ $(1 \leq m \leq l)$. The path with the maximum total trust value of $t\mathrm{TR}_m$ is the selected path as per (6)

$$\begin{cases} \max, & t\mathrm{TR}_m \\ \text{subject to,} & m \in \text{Set } P. \end{cases} \tag{6}$$

It is easy to be shown that (6) is a concave function with a global maximum.

IV. FACT Framework

We make the following assumptions for the vehicular network in which FACT is implemented.

1) Vehicles know their location using global positioning system (GPS) or other localization methods.
2) There are very few (e.g., three) access points deployed in different parts of the city. They are used to propagate system updates and urgent notifications. Traffic information is uploaded into all vehicles once they join the vehicular network. After that and every time they meet an access point, this information is updated.
3) Similar to approaches such as [3], [4], [16], and [19], vehicles transmit beacon periodically to let their neighbors know about their existence. Each beacon has information about the location of the car, its velocity, and the direction it is heading. Using these beacons, each vehicle can determine the neighborhood in which its neighbor is located in order to calculate its trust value.
4) Vehicles are equipped with a temper proof device. Moreover, they can connect to a certificate authority using an access point to receive their certificates. Therefore, they can sign their packets, e.g., source authentication.
5) Whenever there is an incident in the network, such as accident or road problems, two notifications are sent: one at the beginning of that incident and another when the incident is finished. These notifications are called the incident and clearance notifications, respectively.

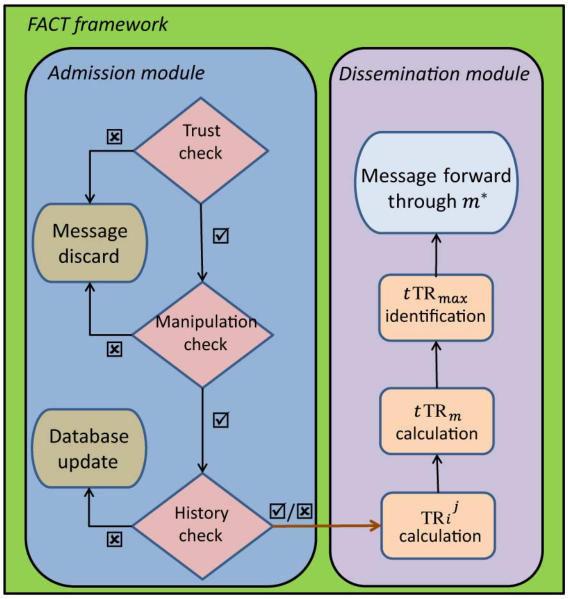


Fig. 2. FACT's framework and how modules function and communicate to each other.

Note that the first four assumptions are typical in any vehicular network. The last assumption increases the overhead; however, this increase is far less than alternative approaches such as sending an acknowledgment for each message. This small increase in overhead is traded to achieve a much better security level.

When a message is received by a vehicle, it has to make sure the message is authentic. FACT is a two-layer framework, which consists of admission and dissemination modules. We divide the vehicle to three types: 1) source, which originates the message; 2) relay, which is in the middle of the path and helps pass the message to its destination(s); and 3) destination that is the last node on the path for which the message is intended. All these types have the FACT framework presented in Fig. 2.

A. Admission Module

When the receiving vehicle is a relay or destination node, the process of authenticating the message happens in the admission module. FACT applies three checks to the message to make sure it is authentic. First of all, the message's origin should be determined. To do that, each node i on the path (including the source) is required to add its location, i.e., its latitude and longitude (x_i, y_i) which can be translated to its road segment and its neighborhood or (r_i, n_i) to the message header. In addition, each node has to also add a time-stamp t_i to the message header showing the current time. Time-stamp and location fields are both assumed to be 4B. x_i and y_i each has 2B allocated in the location field. Since the message size is assumed to be 512B, 8B overhead has a negligible impact on the overall performance of FACT. Using the mentioned fields, when a message is received, the receiver can easily identify the location of the source, e.g., $\mathrm{loc}_s = (r_s, n_s)$. Then, it can use its trust database and based on TR^{r_s} and TR^{n_s}, decide whether it can trust the message's source. FACT uses a simple rule in which a location $\mathrm{loc}_i = (r_i, n_i)$ is considered trusted if the following inequalities hold

$$\begin{cases} \mathrm{TR}^{r_i} > \Phi_r \\ \mathrm{TR}^{n_i} > \Phi_n \end{cases} \tag{7}$$

where Φ_r and Φ_n are acceptable threshold for having a trusted road segment and neighborhood, respectively. The values of Φ_r and Φ_n are tuned by the network administrator based on the application and the characteristics of the environment. The same check is applied to all nodes on the path in addition to the source in which case the message passes the first check only if (7) holds for all previous nodes. Even the current location of the message, i.e., (r_x, n_x) for node x, is evaluated using (7). If both inequalities in (7) hold for (r_x, n_x), then it means that x can trust its current neighborhood. When the message passes the first check, FACT applies the second check.

In the second check, there is a metric that enables FACT to detect any malicious behavior along the path: the time-location pairs (t_i, loc_i) attached to the message where each pair corresponds to one vehicle on the path. If any of the nodes manipulates its location or/and time-stamp in the message to deceive other nodes about the trustworthiness of its (r_i, n_i) pair (and consequently, the values of TR^{r_i} and TR^{n_i}), the next node on the path x can use (t_i, loc_i) for $\{\forall i, i = s, 1, 2, \ldots, x\}$ and easily spot any suspicious information in the (t_i, loc_i) pairs. In fact, x takes advantage of traffic information it has about (r_i, n_i) pairs to come up with an estimate of the time it takes to reach from (r_s, n_s) to (r_x, n_x) through the path traveled by the message, denoted by t_{es}.

To calculate t_{es}, let $\bar{v_i}$ denote the average vehicle speed of the road segment r_i using the historical traffic information. When a message is received by a car x and depending on the implemented forwarding policy and the current vehicle density, it either forwards the message instantaneously (or with a very short-processing delay) or it decides to keep and carry the message for a while, e.g., until it reaches another vehicle. These two cases are called forward and carry phases, respectively. Let t_f denote the transmission time plus the processing delay and $t_c(d)$ denote the time that it takes x to travel d meters. We always have $t_c(d) \gg t_f$. Assume each message goes through k_1 forward and k_2 carry phases until it reaches node x. Vehicle x calculates k_1 and k_2 by looking at $t_i - t_{i-1}$ for $\{\forall i, i = s, 1, 2, \ldots, x\}$ and incrementing k_1 when $t_i - t_{i-1} \leq t_f \pm \delta_1$ and incrementing k_2 when $t_i - t_{i-1} \leq t_c(d) \pm \delta_2$. Parameters δ_1 and δ_2 are tuned based on the channel conditions and the vehicle speed variance, respectively.

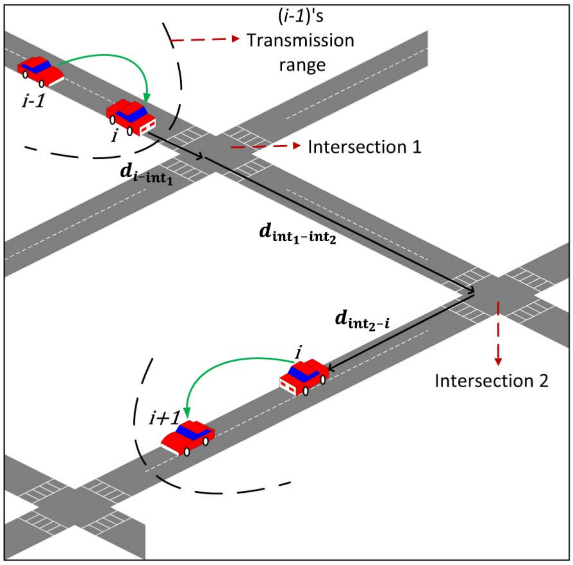


Fig. 3. Example for distance calculation in the carry phase to find $t_c(d)$.

For two successive nodes i and $i+1$, $t_c(d)$ is calculated as given in (8)

$$t_c(d) = \begin{cases} \frac{d_{(i,i)}}{\bar{v_i}}, & \text{if i and i + 1 are on the same road segment} \\ \frac{d_{(i,\mathrm{int}_1)} + d_{(\mathrm{int}_1,\mathrm{int}_2)} + \cdots + d_{(\mathrm{int}_j,i)}}{\bar{v_i}}, & \text{if otherwise} \end{cases} \tag{8}$$

where $d_{(i,i)}$ is the distance that vehicle i traverses until it decides to forward the message to vehicle $i+1$. When i and $i+1$ are on different road segments, vehicle i has to travel until it reaches $i+1$ and on its way, it may cross j intersections. The nominator in the second term in (8) shows the total distance traveled by i before the next transmission. An example of this scenario for $j = 2$ is given in Fig. 3. $d_{(i,\mathrm{int}_1)}$, e.g., is the distance between vehicle i and the intersection int_1. Using (8), x calculates t_{es} as follows:

$$t_{\mathrm{es}} = k_1 t_f + \sum_{k=1}^{k_2} t_c(d_{(k,k+1)}). \tag{9}$$

Then, x compares the calculate estimate with the actual time $t_{\mathrm{ac}} = t_x - t_s$ and if $t_{\mathrm{ac}} - t_{\mathrm{es}} > \Delta$, x concludes that the message is altered on its way. The value of Δ depends on the size of a neighborhood and time it takes to traverse it because if one vehicle decides to manipulate the message, it has to change its location to a more trusted neighborhood different from its current one.

If the message passes the second check as well, x applies the third and final check. In the third check, x takes advantage of the historic knowledge about the location of the source. In fact, this check is a content-based check in which case x validates the event/incident reported in the message. As an example, suppose the message is about an accident at (r_s, n_s). x looks at its knowledge about that location and if r_s is a high-traffic road with prior accidents, it concludes the content can be valid. However, if r_s is a road with very light daily traffic and no history of accidents, it sees an anomaly which may or may not be true. For this purpose, we define an accident threshold ζ to differentiate between a high accident area and a low one. Note that the system administrator can tune this threshold in addition to other system parameters. If x is a relay node, it sets a flag in the message suggesting that the received message is bogus and it cannot be trusted. It then pushes the message down to the FACT's dissemination module to be forwarded to the next node. On the other hand, if x is the destination node, it waits until it receives the same information from multiple sources (three sources for example) then takes an appropriate action, e.g., it uses an alternative route. If another source confirms that incident without the flag being set, because of the contradiction between different sources, x cannot decide whether the message is genuine or not. Therefore, it does not take any action except saving a copy of the message for some time T_{inc}, which represents the average delay between the incident and clearance messages and is set by the network administrator. If during T_{inc}, x receives the clearance notification, meaning that the report was correct, it recalculates the corresponding trust values in its database. Otherwise and when T_{inc} expires, meaning that the report was bogus, x just discards the message. Similarly, when x is a relay node and it realizes that the message was actually correct (using T_{inc}), it updates its knowledge by adding that incident to its database. This means that bogus messages and the corresponding attacks do not hurt the trustworthiness of an area.

B. Dissemination Module

When the receiving vehicle x is a relay node and if the message passes all three checks explained in Section IV-A, x pushes the message down to the dissemination module in order to be forwarded to the next vehicle. Four cases can be imagined for FACT's dissemination module based on the type of the node and its location. Case I: when it is a source node at intersection, it follows the pseudocode given in Algorithm 1. Note that each vehicle knows the location of its neighbors and the corresponding trust values using the periodic beacons. Case II: when it is a relay node at an intersection that wants to decide on the next hop, because the traffic category Cat is already in the message, the relay node follows Algorithm 1 from step 3) to the end. Case III: when the vehicle is a source and in the middle of a street, it takes the first two steps of Algorithm 1 and then forwards the packet to the closest neighbor to the destination. In fact, all vehicles on that segment have the same segment trust value and it does not matter, which vehicle is selected as the next hop. Hence, it is best to send the message to the closest node to the destination. At the end, the sender updates its trust value for that segment. Case V: when the vehicle is a relay node and in the middle of the road segment, it just forwards the message to the closest neighbor to the destination and updates its trust value for that segment.

Note that in situations where there is an adversary node in a trusted area, $\mathrm{ST}_{\mathrm{cur}}$ will be dropped temporarily and so is $t\mathrm{TR}$ for that neighborhood. But, when this node moves out of the trusted area, the total trust will go back to normal. There are cases where all possible paths have very weak total trust values. In those cases, the vehicle carrying the message, regardless of being the source or a relay node, uses a threshold $\mathrm{TR}_{\mathrm{min}}$. If $\exists i$ where $\alpha_i \neq 0$ and $\mathrm{TR}_m^i \leq \mathrm{TR}_{min}$ for $\forall m \in Set\ P$, then the vehicle continues carrying the message until the next intersection to see whether the trust condition changes on that intersection.

Algorithm 1. FACT at Intersection

```
1: Based on the content of the message or the nature of the
   event, the appropriate traffic category, $Cat$, is determined.
2: Based on the selected category, the corresponding $\alpha_i$'s are
   specified.
3: **For set of neighbors,** $j = 1 : w$:
4:    **For set of parameters,** $i = 1 : k$:
5:       $TR_i^j$ is calculated.
6: **For set of paths,** $m = 1 : l$:
7:    $tTR_m$ is calculated using the $\alpha_i$'s calculated in step 2.
8: $tTR_{max}$ is identified.
9: The winning path, $m^*$, is selected and the message includ-
   ing the traffic category, $Cat$, is forwarded to the neighbor
   on $m^*$, i.e., $j^*$.
10: **FOR** $i = 1 : k$:
11:    Based on the forwarding experience, $ST_i^{m^*}$ and $TR_i^{m^*}$
       are updated.
```

There are situations when due to a temporary reason, for instance a construction project or a festival, the trust condition of a road segment or a neighborhood is changed and so is the corresponding trust value in the trust map of vehicles traveling to that area. But when that reason does not exist anymore, if cars use (2), it will take a while for the trust value to converge to the actual value and we say the trust value has a bias. To address this, we consider a parameter, $\mathrm{NUM}_{\mathrm{bias}}$ as the number of times FACT allows the current trust $\mathrm{TR}_{\mathrm{cur}}$ to be hugely different from the historical trust. For example, if $\mathrm{NUM}_{\mathrm{bias}} = 3$ and this is the fourth time that the current trust is much bigger than the historical one, then FACT uses the average of the last three values of $\mathrm{TR}_{\mathrm{cur}}$ as the trust value for that segment. The size of the gap between the current and the historical trust, i.e., the size of the bias, should be carefully selected to avoid erroneous decisions. One example could be 50% of the historical trust.

One limitation of FACT is when a malicious vehicle travels in a known-to-be-trusted segment. Therefore, all messages that are relayed to this vehicle will be either manipulated or destroyed. The other vehicles receive a very bad feedback and they try to update their trust estimate for this segment. Using the above-mentioned mechanism and assuming the malicious vehicle corrupts the packets more than $\mathrm{NUM}_{\mathrm{bias}}$ times, the trust value will be declined dramatically. But, as the malicious node exits this segment, the trust value for that segment goes back to normal after at most $\mathrm{NUM}_{\mathrm{bias}}$ times of genuine transmissions. Based on this discussion, the impact of infiltrating a malicious node in a trusted segment can be minimized by choosing a small value for $\mathrm{NUM}_{\mathrm{bias}}$, so that both discovery and recovery periods are small.

Finally, it is obvious that the same satisfaction value caused by a car experiment calculated via (5) will affect the trust value of the segment as well as the neighborhood, as per (2) and (4). However, the effect of the current value should have less influence on the neighborhood compare to the segment. Therefore, the system administrator, which sets the parameters of the vehicle, may set the values of γ and η of the neighborhood less than the parameters γ and η of a segment in that neighborhood. There is a similar situation when setting the values of τ and n in (4) and τ and v in (2), where the administrator needs to be careful in choosing the appropriate values for the segment and for the neighborhood.

V. Analysis

In this section, we analyze our proposed framework from the security point of view and then present a qualitative comparison between FACT and other trust-based approaches.

A. Security Analysis

We follow the well-known Dolev–Yao approach [28] and provide an adversary analysis. Dolev–Yao assume that all packets are delivered to and received from an adversary, which is capable of recording, deleting, replaying, rerouting, reordering, and rescheduling the packets. To attack this system, an adversary can be a sender or a relay node. Therefore, we provide three adversary models representing three positions of the adversary. In our first model, our adversary is a sender, and in second and third models, our adversary is a relay node.

1) First Adversary Model: In this model, we assume our adversary is a sender. To be more precise, he sends the bogus messages about, e.g., traffic.

Objective: The objective of the adversary can be misleading the receivers about the traffic in an area (e.g., street), and/or defecting a neighborhood/segment.

Initial capability: Initially, he knows the detailed information about our proposed framework as well as the overall topology and map of the area.

Capability during the attack: He is capable to send many bogus massages about an event that never happened. Moreover, he is capable of modifying the source of the messages to be other nodes, a benign/valid car, or a dummy car. Also, he can keep driving and move from one area (neighborhood/segment) to another area to harm the trust values of multiple areas.

Discussion: As per our design, explained part in Section IV-B, having $\mathrm{NUM}_{\mathrm{bias}}$, a receiver wait for $\mathrm{NUM}_{\mathrm{bias}}$ reports about the same event to rely on the report. This value can be tuned up if necessary, if the system get under attack, especially distributed attack. Note that distributed attack in this case means our adversary should have multiple cars in different street and areas to be able to completely destroy the trust evaluation and misleads the receivers, which is a very expensive attack and may not be feasible.

Finally, since the adversary is the packet generator, a receiver may get mislead for the first time; however, after a few catching the errors and finding out about the bogusness of the packets, the receiver can identify that particular sender as a malicious/attacker and denies entire information generated by the sender from that point.

Note that if our adversary generates the bogus messages and modifies the sender of each one to be a benign/valid car or a dummy car to perform the attack, it still may affect the trust value of an area. It is obvious that if he tries to impersonates a benign car, the benign car will receive the message as well, as a receiver per our design, and will distribute a correction message to disregard the bogus one. In case of impersonating a dummy car, as per our discussion in Section IV-A, our proposed solution catches and ignores that attacks.

2) Second Adversary Model: In this model, we assume our adversary is a relay node. To be more precise, he does not generate any packet, and instead, he only relies the packet sent by pother while modifying them.

Objective: The objective of the adversary can be destroying/ modifying the senders messages about an event and about an area.

Initial capability: Initially, he knows the detailed information about our proposed framework and also he is well equipped to receive and then sends the packets as well as showing himself as a ready node to relay.

Capability during the attack: He is capable of keep receiving and sending the messages. He can read inside the packet and can modify any part of the message, like the time stamp, street name, and overall the event. For instance, he can change a message instead of saying “there is an accident,” he can modify it to “the accident is cleared/traffic is smooth.” He also can receive the message and drops it, or deliver it to a wrong direction, in case of unicast/multicast communication. In this case, he may modify the destination as well to mislead other relay nodes that will receive this message, which is passing the adversary node.

Discussion: First of all, in our design, we assumed there is an under-layer security mechanism such that the integrity of the messages and packets are preserved. In this case, when the affected/modified packet are received by the next relay node/ or a receiver (if the adversary is the last node before the receiver), the receiver and the next relay nodes are capable of catching the attack as part of their admission module of the FACT framework (Section IV-A).

In case of the next node being a receiver, the receiver does not rely on packet received from the adversary anymore and expects the message from others.

In the case where the next node being a relay node, still the next relay node is capable of finding the attack and therefore easily can stop cooperating with the adversary for the following communication.

Note that, since each relay node should adds a time stamp to the packets, if the adversary attacks the system by more than one car, meaning damages the packet by first car, and passing it to the next attacker car to relay, still along the path, or in worth case by a receiver, the attack can be discovered. In either cases, the benign node(s) can stop their cooperation with the packets passing the attackers.

As per Section IV-B by having $\mathrm{NUM}_{\mathrm{bias}}$, a receiver waits for $\mathrm{NUM}_{\mathrm{bias}}$ number of the reports about the same event to rely on the report. This value can be tuned up if necessary, if the system get under attack, especially distributed attack. Note that distributed attack in this case means our adversary should have

TABLE I
COMPARING THE FEATURES OF THE SURVEYED PROPOSALS IN SECTION II AND OUR PROPOSAL

Item/ paper	Data-centric	Entity-centric	Source-centric	Receiver-centric	Path-centric	Infrastructure-based	Application-oriented	Broadcast/ unicast
[6]	✖	✔	✔	✖	✖	✔	✖	B
[7]	✔	✖	✖	✔	✖	✔	✖	B
[9]	✖	✖	✖	✖	✔	✔	✖	U
[10]	✖	✔	✔	✖	✔	✔	✖	U
[11]	✖	✔	✔	✖	✖	✖	✖	B
[12]	✖	✔	✔	✖	✖	✖	✖	B
[13]	✔	✔	✔	✖	✔	✖	✖	B
[14]	✔	✖	✖	✔	✖	✖	✖	B
[15]	✔	✔	✖	✔	✖	✖	✖	B
[16]	✔	✖	✖	✔	✖	✔	✖	B
[17]	✖	✔	✔	✖	✖	✖	✖	B
[18]	✖	✔	✔	✖	✖	✖	✖	B
[19]	✔	✔	✔	✖	✖	✔	✖	U
[20]	✖	✖	✖	✖	✔	✖	✖	U
[21]	✔	✔	✔	✖	✖	✖	✖	U
[22]	✖	✔	✔	✔	✖	✖	✖	U
[23]	✖	✔	✔	✖	✔	✖	✖	U
[24]	✔	✖	✔	✖	✔	✖	✖	U
[25]	✔	✔	✔	✖	✔	✖	✖	U
FACT	✔	✔	✔	✔	✔	✖	✔	B and U

multiple cars in different street and areas to be able to completely destroy the trust evaluation and misleads the receivers, which is a very expensive attack and may not be feasible.

Finally, since the adversary is the packet generator, a receiver may get mislead for the first time; however, after a few catching the errors and finding out about the bogusness of the packets, the receiver can identify that particular sender as a malicious/attacker and denies entire information generated by the sender from that point.

3) Third adversary model: In this model, we assume our adversary is again a relay node. However, unlike the second model, he does not change the messages; instead, he forwards the messages along a wrong path (with low-trust value).

Objective: The objective of the adversary can be disaffecting the messages about an event. The message can be about an accident happening or about clearing the traffic caused by an accident. With a good approximation, the former one is similar to a good-mouthing attack, and later one is similar to a bad-mouthing attacks.

Initial capability: Initially, he knows the detailed information about our proposed framework as well as the overall topology and map of the area.

Capability during the attack: He is capable of playing the role of a relay node. He forwards the messages, but not to the proper direction. Therefore, he keeps his reputation to others as a benign node although he is performing the attack. Note that this attack can be similar to the black-hole attack in a unicast communication scenario where the relay node sends the packets to a malicious node to be dropped at that node.

Discussion: As per our previous discussion, our application is data dissemination. Therefore, we do not have a single receiver (like in the unicast communication scenario). In fact, the messages are being forwarded and disseminated by other cars, as well. So, if an attacker tries to send the messages to a path with low-trust value, to perform the attack, other benign cars will redirect the messages to the proper path (with high-trust value) to be disseminated.

B. Comparison Analysis

Table I presents a brief summary of the comparison between the studied models in Section II and our proposal. Rows of the table show the characteristics of the mechanism/systems, as follows.

1) *Data-centric:* If the mechanism/system concentrates on the content of the packet.
2) *Entity-centric:* If the mechanism/system concentrates on the entity, e.g., sender, receiver, and/or a relay node/vehicle.
3) *Source-centric:* If the mechanism/system is designed to be used by the senders, e.g., for choosing the next hop or the best path.
4) *Receiver-centric:* If the mechanism/system is designed to be used by the receivers, e.g., for trusting the received message.
5) *Path-centric:* If the mechanism/system is designed to be used for choosing a path or evaluating the trustworthiness of the path.
6) *Infrastructure-based:* If the mechanism/system requires having an infrastructure, including road side unit or even certificate authority.
7) *Application-oriented:* If the mechanism/system considers requirements of the application that is running.
8) *Broadcast/unicast:* If the mechanism/system uses broadcasting or unicasting for communicating between vehicles.

FACT is data-centric because it checks the content of the message once for determining traffic category and again in the third check of the admission module. It is entity-centric simply because FACT assures that the message can be trusted only

TABLE II
CONFIGURATION PARAMETERS

Parameter	Value
Channel modeling	Rician with $K = 10$
Packet length (B)	200
MAC standard	IEEE 802.11p
Communication frequency	5.9 GHz
Data rate (Mbit/s)	6
Data traffic	Periodic with $T = 50$ ms
Slot duration	13 μs
SIFS	32 μs
DIFS	58 μs
Minimum vehicle speed (km/h)	80
Maximum vehicle speed (km/h)	130
Average vehicle speed (km/h)	100

when all the nodes (entities) on the path are trustworthy (at least to some extent). FACT is both source-centric and receiver-centric because each receiving car first makes sure the received message can be trusted (in admission module), and then it chooses a trusted path for forwarding the message (in dissemination module). Using the same reasoning, it is easy to see that FACT is path-centric, as well. Since having road-side units are not necessary for FACT to work, it is not infrastructure-based. It is application-oriented because it specifies α_i's based on the application. Finally, both broadcast and unicast are supported by FACT.

VI. PERFORMANCE EVALUATION

To evaluate the performance of the FACT, we ran an extensive set of simulations using MATLAB. We divided the whole city into three regions. 1) Very trusted neighborhood N_t where 90% of transmissions are genuine while 10% of times packets are destroyed. The size of N_t, on the other hand is bigger than the other two regions and it has a lower vehicle density resulting in a bigger delay. 2) Untrusted neighborhood N_u where only 10% of transmissions are genuine. However, N_u has a smaller size and higher vehicle density compared to other two regions, which means a smaller delivery delay. 3) Semitrusted neighborhood N_s where 50% of transmissions are genuine. The size and the vehicle density in N_s are between the respective values in N_t and N_u, which results in a delay value between other two regions' delays. The application is assumed to be a safety-related service with $\alpha_i = 1/3$ for delay, reliability, and data integrity. We assumed a transmission range of 250 m and Rician fading with shadowing as the propagation model. The velocity of cars is assumed to follow a normal distribution according to work by Wisitpongphan *et al.* [29] using real traffic traces. Simulation parameters are summarized in Table II.

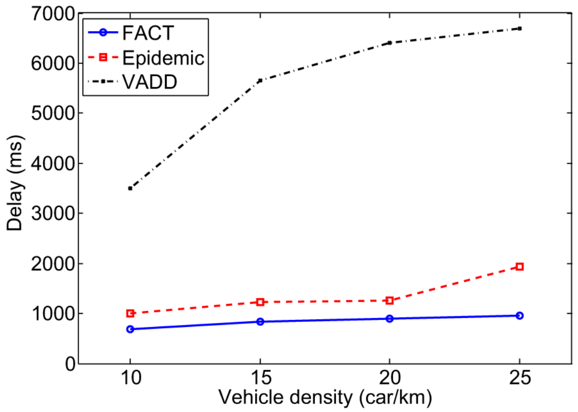


Fig. 4. Delivery delay versus different vehicle densities for 90% reliability.

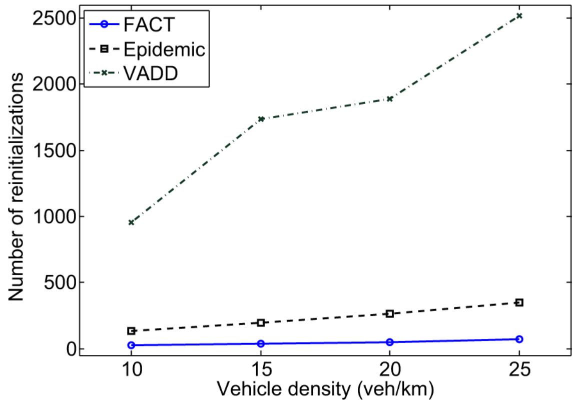


Fig. 5. Number of times the source retransmits its message because of the packet drops by malicious nodes.

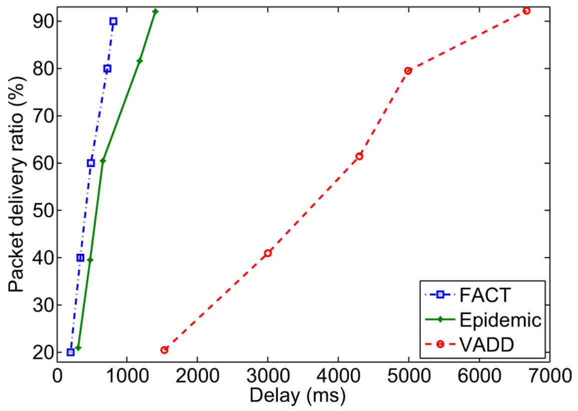


Fig. 6. Packet delivery ratio versus delivery delay.

The results of the simulations for delay are reported in Fig. 4 for FACT versus epidemic routing and VADD. As shown in this figure, FACT outperforms other two algorithms in terms of delay while achieving the same level of reliability although they use faster paths. FACT performance versus VADD is particularly interesting because even though VADD chooses the fastest path and benefits half of FACT's communication delay per transmission, delivery delay in FACT is significantly improved (between 400% and 600%). The reason is the excessive number of retransmissions in VADD as a result of packet drops in the untrusted neighborhood (see Fig. 5). Each time a packet is dropped by a malicious vehicle, the source vehicle has to resend the original message, which leads to higher delivery delay.

On the other hand, packet delivery ratio (or the end-to-end reliability) is the other victim of untrusted vehicles. As shown in Fig. 6, FACT jumps to maximum packet delivery ratio in a much shorter time compared to epidemic routing and particularly compared to VADD. It means that FACT achieves higher reliability in a shorter time. Note that although epidemic routing is a close competitor to FACT in these figures, but

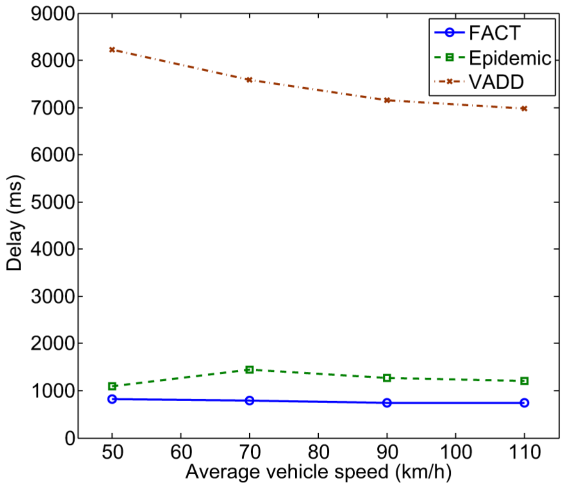


Fig. 7. Delivery delay versus different average vehicle speeds for 90% reliability.

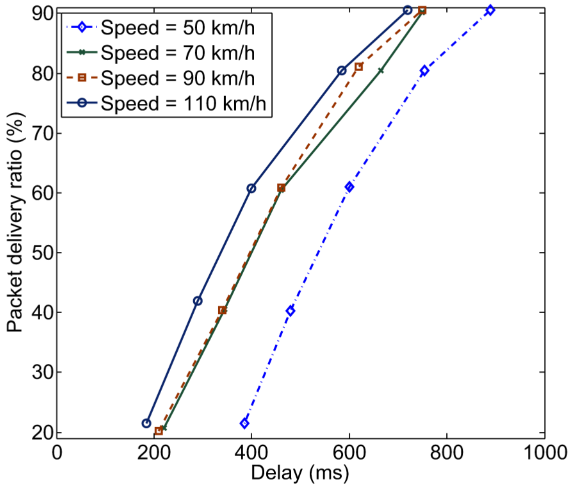


Fig. 8. FACT packet delivery ratio versus delivery delay for different average speeds.

because of the flooding nature of it, epidemic routing imposes a huge communication cost in terms of the interference and congestion to the network. However, FACT has a communication cost comparable to VADD, which is much less than epidemic.

Next, we study the effect of vehicle speed on delivery delay. A comparison between FACT and VADD and epidemic routing in terms of delivery delay against different speeds is presented in Fig. 7. The average vehicle speed ranges from 50 km/h, which is typical for an urban scenario to 110 km/h, which is more suited for a highway environment. As shown in this figure, vehicle speed's impact on delivery delay is very high for VADD and very low for FACT. As expected, increasing the speed reduces the delivery delay for all three algorithms.

Finally, the impact of vehicle speed on the FACT packet delivery ratio is studied. As shown in Fig. 8, there is a fairly big gap between when the speed is 50 km/h and the rest of speeds. It seems 50 km/h is a transitional speed in this regards. The results for speed of 70 and 90 km/h are very close to each other. The next transitional speed seems to be 110 km/h.

VII. Conclusion

In this paper, first we present a survey on routing, security and trust systems/mechanisms, mainly in vehicular networks and *ad hoc* environment. Then, we propose a two-layer trust-based information dissemination framework called FACT and its application in vehicular networks. FACT is an application-oriented framework designed to support broadcast, multicast, and unicast communication in vehicular networks. FACT maintains the trust values of the neighborhoods and segments of the city. It first makes sure the message is originated from and traveled through trusted nodes and the content is valid. It then uses stored trust values to choose the best path to route the message, or if needed, to carry the messages. Simulation results show that FACT outperforms other well-known routing protocols like VADD. They also validate the effectiveness (in terms of communication cost) and scalability of FACT. FACT gives network designers a full package, which delivers trustworthy messages through a safe path with high reliability and in a short amount of time. It is flexible enough meaning network admins can tune the parameters based on the network condition. Designers can incorporate their desired scheduling and routing schemes into FACT and still disseminate the messages safely. FACT is a framework that supports different applications with different requirements.

References

[1] K. Rostamzadeh and S. Gopalakrishnan, "Analysis of message dissemination in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 62, no. 8, pp. 3974–3982, Oct. 2013.

[2] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Duke Univ., Durham, NC, USA, Tech. Rep. CS-200006, 2000.

[3] J. Zhao and G. Cao, "VADD: Vehicle-assisted data delivery in vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 57, no. 3, pp. 1910–1922, May 2008.

[4] A. Skordylis and N. Trigoni, "Delay-bounded routing in vehicular ad-hoc networks," in *Proc. MobiHoc*, 2008, pp. 341–350.

[5] J.-H. Cho, A. Swami, and R. Chen, "A survey on trust management for mobile ad hoc networks," *IEEE Commun. Surveys Tuts.*, vol. 13, no. 4, pp. 562–583, 2011.

[6] D. Huang, X. Hong, and M. Gerla, "Situation-aware trust architecture for vehicular networks," *IEEE Commun. Mag.*, vol. 48, no. 11, pp. 128–135, Nov. 2010.

[7] M. Raya, P. Papadimitratos, V. Gligor, and J.-P. Hubaux, "On data-centric trust establishment in ephemeral ad hoc networks," in *Proc. IEEE 27th Conf. Comput. Commun. (INFOCOM'08)*, 2008, pp. 1238–1246.

[8] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The eigentrust algorithm for reputation management in p2p networks," in *Proc. 12th Int. Conf. World Wide Web*, 2003, pp. 640–651.

[9] A. Koster *et al.*, "Using trust and possibilistic reasoning to deal with untrustworthy communication in VANETs," in *Proc. 16th Int. IEEE Annu. Conf. Intell. Transp. Syst.*, The Hague, The Netherlands, Oct. 6–9, 2013, pp. 2355–2360.

[10] H. Zhu, S. Du, Z. Gao, M. Dong, and Z. Cao, "A probabilistic misbehavior detection scheme toward efficient trust establishment in delay-tolerant networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 1, pp. 22–32, Jan. 2014.

[11] X. Li, Z. Jia, P. Zhang, R. Zhang, and H. Wang, "Trust-based on-demand multipath routing in mobile ad hoc networks," *IET Inf. Secur.*, vol. 4, no. 4, pp. 212–232, Dec. 2010.

[12] S. Sarkar and R. Datta, "A trust based protocol for energy-efficient routing in self-organized manets," in *Proc. Annu. IEEE India Conf. (INDICON)*, 2012, pp. 1084–1089.

[13] T. Gazdar, A. Rachedi, A. Benslimane, and A. Belghith, "A distributed advanced analytical trust model for VANETs," in *Proc. Global Commun. Conf. (GLOBECOM)*, 2012, pp. 201–206.

[14] F. Gómez Mármol and G. Martínez Pérez, “Trip, a trust and reputation infrastructure-based proposal for vehicular ad hoc networks,” *J. Netw. Comput. Appl.*, vol. 35, no. 3, pp. 934–941, 2012.

[15] F. Dotzer, L. Fischer, and P. Magiera, “VARS: A vehicle ad-hoc network reputation system,” in *Proc. 6th IEEE Int. Symp. World Wireless Mobile Multimedia Netw. (WoWMoM’05)*, 2005, pp. 454–456.

[16] D. Tian, Y. Wang, H. Liu, and X. Zhang, “A trusted multi-hop broadcasting protocol for vehicular ad hoc networks,” in *Proc. Int. Conf. Connect. Veh. Expo. (ICCVE)*, 2012, pp. 18–22.

[17] J. Wang, Y. Liu, X. Liu, and J. Zhang, “A trust propagation scheme in VANETs,” in *Proc. IEEE Intell. Veh. Symp.*, 2009, pp. 1067–1071.

[18] D. Huang, S. A. Williams, and S. Shere, “Cheater detection in vehicular networks,” in *Proc. IEEE 11th Int. Conf. Trust Secur. Privacy Comput. Commun. (TrustCom)*, 2012, pp. 193–200.

[19] Y.-C. Wei and Y.-M. Chen, “An efficient trust management system for balancing the safety and location privacy in VANETs,” in *Proc. IEEE 11th Int. Conf. Trust Secur. Privacy Comput. Commun. (TrustCom)*, 2012, pp. 393–400.

[20] M. H. Eiza and Q. Ni, “A reliability-based routing scheme for vehicular ad hoc networks (VANETs) on highways,” in *Proc. IEEE 11th Int. Conf. Trust Secur. Privacy Comput. Commun. (TrustCom)*, 2012, pp. 1578–1585.

[21] M. Teler and V. Cristea, “Securing vehicular networks using deterministic schemes for computing trust,” in *Proc. IEEE 4th Int. Conf. Intell. Netw. Collab. Syst. (INCoS)*, 2012, pp. 214–221.

[22] M.-C. Chuang and J.-F. Lee, “TEAM: Trust-extended authentication mechanism for vehicular ad hoc networks,” *IEEE J. Syst.* vol. 8, no. 3, pp. 749–758, Jan. 2013.

[23] T. Gazdar, A. Rachedi, A. Benslimane, and A. Belghith, “A distributed advanced analytical trust model for VANETs,” in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, 2012, pp. 201–206.

[24] N. Haddadou and A. Rachedi, “DTM^2: Adapting job market signaling for distributed trust management in vehicular ad hoc networks,” in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2013, pp. 1827–1832.

[25] C. Liao, J. Chang, I. Lee, and K. K. Venkatasubramanian, “A trust model for vehicular network-based incident reports,” in *Proc. IEEE 5th Int. Symp. Wireless Veh. Commun. (WiVeC)*, 2013, pp. 1–5.

[26] S. Buchegger and J.-Y. Le Boudec, “Performance analysis of the CONFIDANT protocol,” in *Proc. 3rd ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2002, pp. 226–236.

[27] A. Das and M. M. Islam, “Securedtrust: A dynamic trust computation model for secured communication in multiagent systems,” *IEEE Trans. Dependab. Secure Comput.*, vol. 9, no. 2, pp. 261–274, Mar./Apr. 2012.

[28] D. Dolev and A. Yao, “On the security of public key protocols,” *IEEE Trans. Inf. Theory*, vol. 29, no. 2, pp. 198–208, Mar. 1983.

[29] N. Wisitpongphan, F. Bai, P. Mudalige, V. Sadekar, and O. Tonguz, “Routing in sparse vehicular ad hoc wireless networks,” *IEEE J. Sel. Areas Commun.*, vol. 25, no. 8, pp. 1538–1556, Oct. 2007.

Karim Rostamzadeh (GSM’09) received the B.Sc. and M.Sc. degrees in electrical engineering from the Isfahan University of Technology, Khomeynishahr, Iran, in 2006 and 2008, respectively, and is currently working toward the Ph.D. degree at the University of British Columbia, Vancouver, BC, Canada.

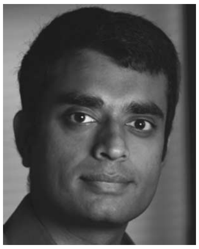
In 2009, he joined the Department of Electrical and Computer Engineering, University of British Columbia. His research interests include scheduling and routing in wireless communication networks, providing services to traveling vehicles under intelligent transportation systems (ITS) umbrella, and modeling delay and reliability in information dissemination in mobile networks.

Hasen Nicanfar (S’11) received the BA.Sc. degree in electrical engineering from the Sharif University of Technology, Tehran, Iran, in 1993, the MA.Sc. degree in computer networks from Ryerson University, Toronto, ON, Canada, in 2011, and is currently working toward the Ph.D. degree in electrical and computer engineering at the University of British Columbia, Vancouver, BC, Canada.

From 1993 to 2010, he worked in different professional capacities such as IT/ERP Manager, Project Manager, and Business and System Analyst. His research interests include trust, security and privacy in wireless communication, computer network, and cloud computing.

Narjes Torabi (GSM’10) received the B.Sc. and M.Sc. degrees in computer engineering from the Isfahan University of Technology, Isfahan, Iran, in 2006 and 2008, respectively, and is currently working toward the Ph.D. degree in electrical and computer engineering at the University of British Columbia, Vancouver, BC, Canada.

In 2009, she joined the Department of Electrical and Computer Engineering, University of British Columbia. Her research interests include the design of coexistence management, medium access control and scheduling schemes, and realization and implementation of public m-health service.



Sathish Gopalakrishnan (S’02–M’06) received the M.S. degree in applied mathematics and Ph.D. degree in computer science from the University of Illinois at Urbana-Champaign, Champaign, IL, USA, in 2004 and 2005, respectively.

In 2007, he joined the University of British Columbia (UBC), Vancouver, BC, Canada, where he is currently an Associate Professor of electrical and computer engineering. His research interests include center around resource allocation problems in several contexts including real-time, embedded systems, and wireless networks.

Dr. Gopalakrishnan was the recipient of awards for his work from the IEEE Industrial Electronics Society (Best Paper in the IEEE Transactions on Industrial Informatics in 2008) and at the IEEE Real-Time Systems Symposium (in 2004).

Victor C. M. Leung (S’75–F’03) received the B.A.Sc. (Hons.) and Ph.D. degrees in electrical engineering from the University of British Columbia (UBC), Vancouver, BC, Canada, in 1977 and 1981, respectively.

He attended the Graduate School, UBC, on a Natural Sciences and Engineering Research Council Postgraduate Scholarship. From 1981 to 1987, he was a Senior Member of Technical Staff and Satellite System Specialist with MPR Teltech Ltd., Vancouver, BC, Canada. In 1988, he was a Lecturer with the Department of Electronics, Chinese University of Hong Kong, Shatin, Hong Kong. He returned to UBC as a Faculty Member in 1989, and is currently a Professor and the TELUS Mobility Research Chair of Advanced Telecommunications Engineering with the Department of Electrical and Computer Engineering. He has coauthored more than 750 technical papers in international journals and conference proceedings, 29 book chapters, and has coedited 8 books. His research interests include the wireless networks and mobile systems.

Dr. Leung is a Registered Professional Engineer in the Province of British Columbia, Canada. He is with the Royal Society of Canada, the Engineering Institute of Canada, and the Canadian Academy of Engineering. He was a Distinguished Lecturer of the IEEE Communications Society. He is a Member of the Editorial Board of the IEEE Wireless Communications Letters, *Computer Communications*, and several other journals, and has previously served on the Editorial Board of the IEEE Journal on Selected Areas in Communications, the IEEE Transactions on Wireless Communications, the IEEE Transactions on Vehicular Technology, the IEEE Transactions on Computers, and the *Journal of Communications and Networks*. He has guest-edited many journal special issues, and contributed to the Organizing Committees and Technical Program Committees of numerous conferences and workshops. He was a recipient of the IEEE Vancouver Section Centennial Award and the 2012 UBC Killam Research Prize. He was the recipient of the APEBC Gold Medal as the head of the graduating class in the Faculty of Applied Science, UBC. Several of his papers have been selected for Best Paper Awards.